15-441
15-641 Computer Networking

Lecture 18: Delivering Content
Content Delivery Networks
Peter Steenkiste

Fall 2014
www.cs.cmu.edu/~prs/15-441-F14

Overview

- Web
- Consistent hashing
- Peer-to-peer
- CDN
  - Motivation
  - Edge servers
  - Content delivery
  - Mapping
  - Impact on Internet

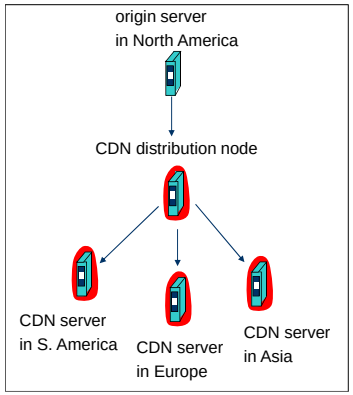
Some slides based on presentation by Patrick Gilmore

Content Distribution Networks (CDNs)

- The content providers are the CDN customers.

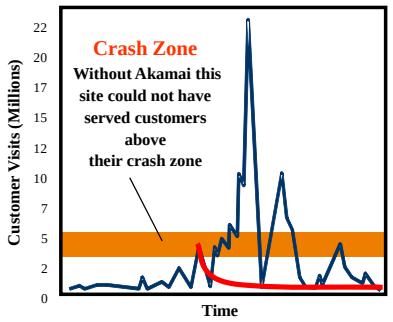
Content replication

- CDN company installs hundreds of CDN servers throughout Internet
  - Close to users
- CDN replicates its customers' content in CDN servers. When provider updates content, CDN updates servers



http://www.akamai.com/html/technology/nui/news/index.html

Case Study on Reliability and Scalability: The 2000 Election



What is the CDN?

- Edge Caches: work with ISP and networks everywhere to install edge caches
  - Edge = close to customers
- Content delivery: getting content to the edge caches
  - Content can be objects, video, or entire web sites
- Mapping: find the "closest" edge server for each user and deliver content from that server
  - Network proximity not the same as geographic proximity
  - Focus is on performance as observed by user (quality)

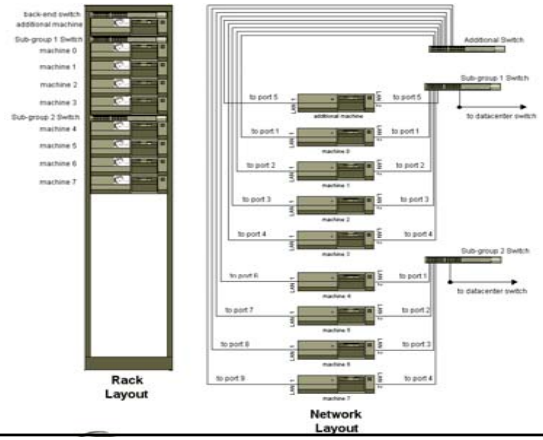
Potential Benefits

- Very good scalability
  - Near infinite if deployed properly
- Good economies at large scales
  - Infrastructure is shared efficiently by customers
  - Statistical multiplexing: hot sites use more resources
- Avoids congestion and long latencies
  - Through mapping to closest server
- Can be extremely reliable
  - Very high degree of redundancy
  - Can mitigate some DoS attacks

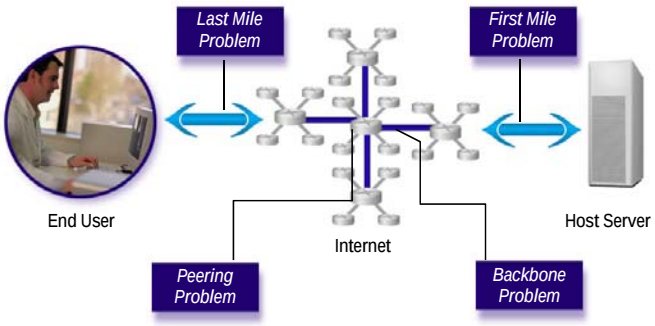
Edge Caches

- Region – set of caches managed as a cluster
  - May have a specific function: http, streaming, …
- Availability is a major concern in architecture design
- Redundancy at the network level
  - See next slide
- Dealing with server failures
  - Servers do fail occasionally
  - Each server has a "buddy" which is constantly trading hellos
  - If hellos stop, buddy starts to respond directly to requests for primary server
  - Users in the middle of a download may have to hit "reload"
  - No one else will notice any interruption

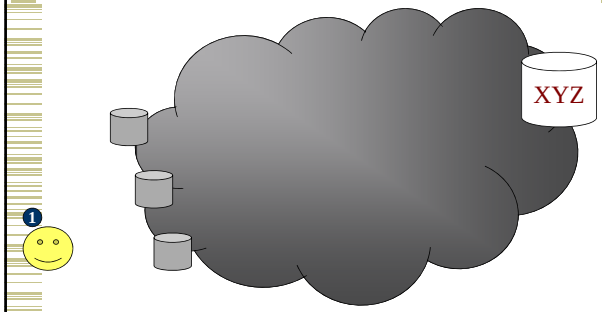
Example Configuration



Content Delivery: Possible Bottlenecks

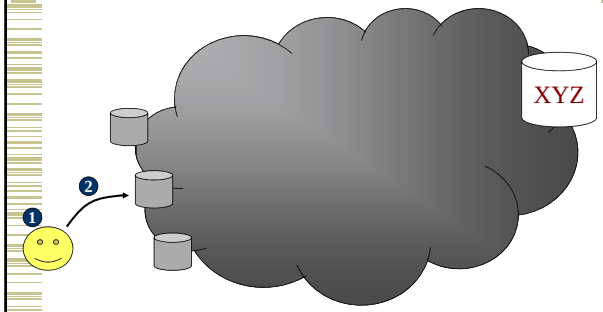


Process Flow



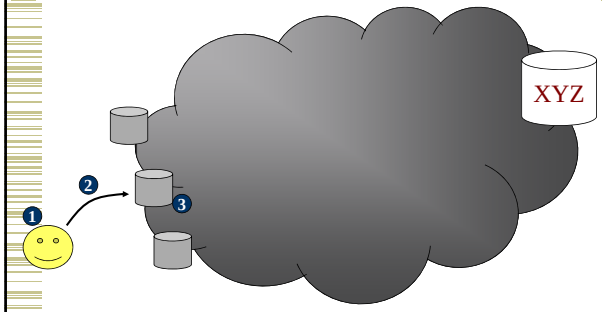
1. User wants to download distributed web content

Process Flow



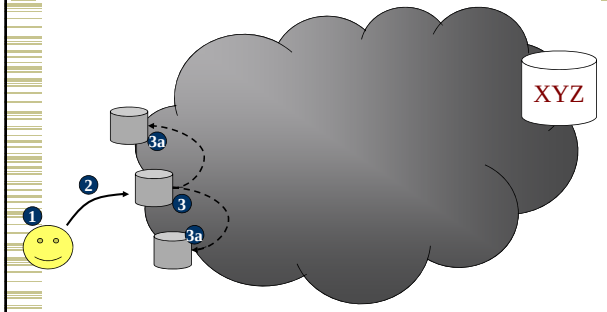
2. User is directed through Akamai's dynamic mapping to the "closest" edge cache

Process Flow



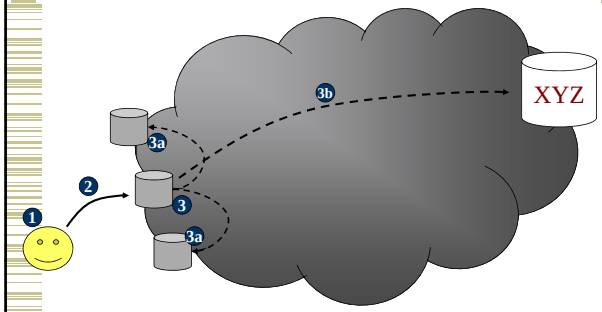
3. Edge cache searches local hard drive for content

Process Flow



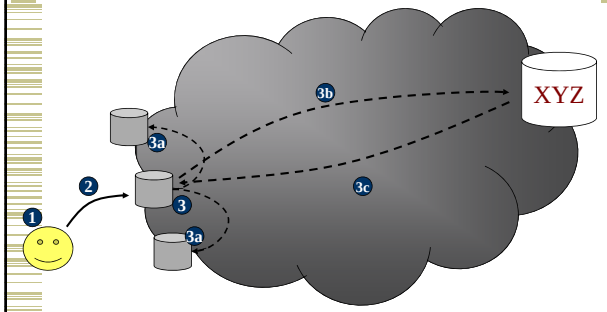
3b. If requested object is not on local hard drive, edge cache checks other edge caches in same region for object

Process Flow



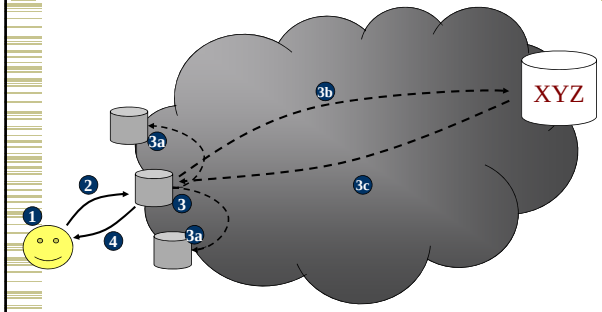
3b. If requested object is not cached or not fresh, edge cache sends an HTTP GET the origin server

Process Flow



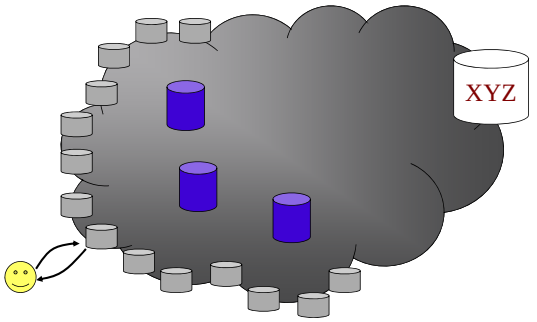
3c. Origin server delivers object to edge cache over optimized connection

Process Flow



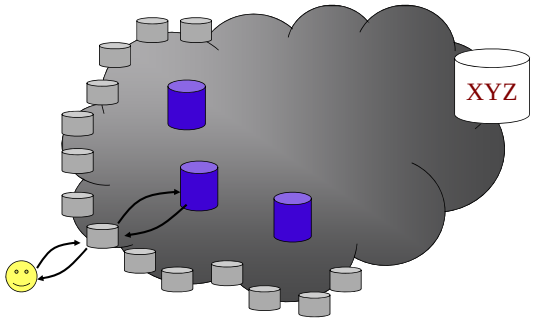
4. Edge server delivers content to end user

Core Hierarchy Regions



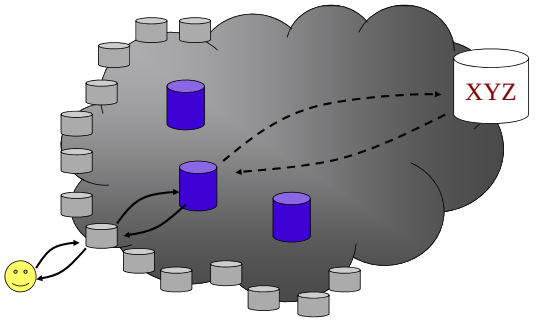
1. User requests content and is mapped to optimal edge Akamai server

Core Hierarchy Regions



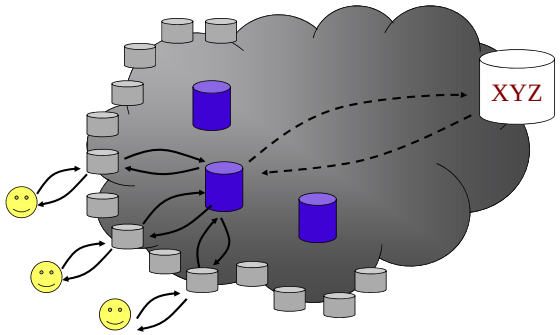
2. If content is not present in the region, it is requested from most optimal core region

Core Hierarchy Regions



3. Core region makes one request back to origin server

Core Hierarchy Regions



4. Core region can serve many edge regions with one request to origin server

Core Hierarchy Features

Reduces traffic back to origin server

- Reduces infrastructure needs of customer
- Provides best protection against flash crowds
  - Especially important for large files (*e.g.* Operating System updates or video files)

Improved end-user response time

- Core regions are well connected
- Optimized connection speeds object delivery

Mapping: Server Selection

- Which server?
  - Lowest load → to balance load on servers
  - Best performance → to improve client performance
    - Based on Geography? RTT? Throughput? Load?
  - Any alive node → to provide fault tolerance
- How to direct clients to a particular server?
  - As part of routing → anycast, cluster load balancing
    - Not covered ☹
  - As part of application → HTTP redirect
  - As part of naming → DNS

Application Based

- HTTP supports simple way to indicate that Web page has moved (30X responses)
- Server receives Get request from client
  - Decides which server is best suited for particular client and object
  - Returns HTTP redirect to that server
- Can make informed application specific decision
- May introduce additional overhead → multiple connection setup, name lookups, etc.
- While good solution in general, but…
  - HTTP Redirect has some design flaws

Naming Based

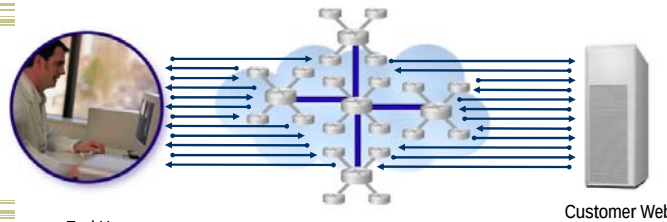
- Client does name lookup for service
- Name server chooses appropriate server address
  - A-record returned is “best” one for the client
- What information can name server base decision on?
  - Server load/location → must be collected
  - Information in the name lookup request
    - Name service client → typically the local name server for client

25

Mapping Algorithms

- Three main components to finding “closest” edge cache to end user from a Network point of view:
  - Packet Loss + Throughput + Latency
  - Listed in order of importance (roughly)
- Mapping also takes into account edge cache performance
  - Does a server have an object on its hard drive?
  - Uses consistent hashing algorithm
  - Does the edge cache have CPU, RAM, bandwidth, etc. available to serve end-user?

Access to Web Site without CDN



1 User enters standard URL www.customer.com

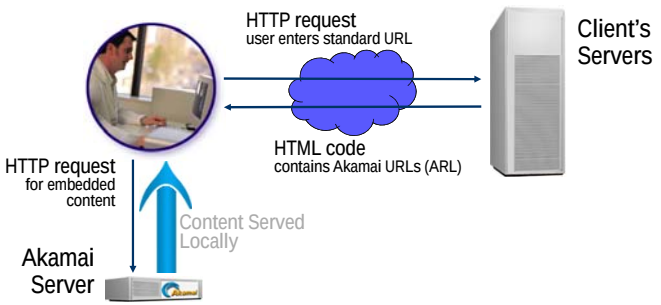
2 Customer’s Web Server returns HTML with embedded URLs

3 User’s browser requests embedded objects from customer Web server

4 Objects served with round trips across the Internet

<img src=/images/logo.gif>

FreeFlow – Akamai’s Object Delivery Service



Example ARL: img src= a1000.g.akamai.net/…/www.customer.com/images/logo.gif

Steps in Content Retrieval

- Clients fetch html document from primary server
  - E.g. fetch index.html from cnn.com
- URLs for replicated content are replaced in html
  - E.g. <img src="http://cnn.com/af/x.gif"> replaced with <img src="http://a73.g.akamaitech.net/7/23/cnn.com/af/x.gif">
  - Note that modified name includes original file name
  - Drawback: must modify the content
- Client is forced to resolve aXYZ.g.akamaitech.net hostname
- Focus is on static content
  - Can be freely cached

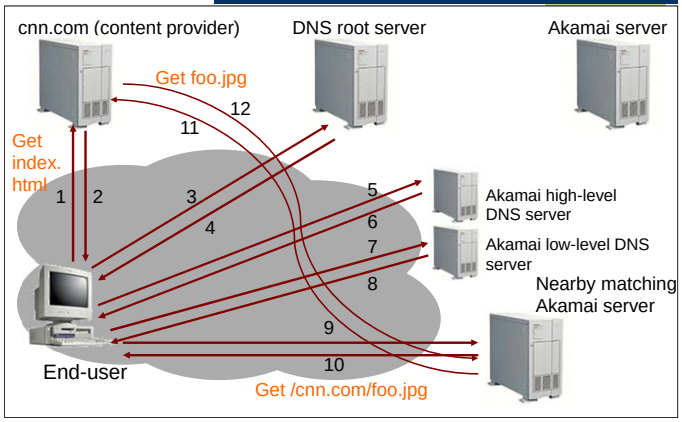
29

Resolution of Modified Name

- Root server gives NS record for akamai.net
- Akamai.net name server returns NS record for g.akamaitech.net
  - Name server chosen to be in region of client's name server
  - TTL is large
- G.akamaitech.net nameserver chooses server in region
  - Should try to chose server that has file in cache - How to choose?
  - Uses aXYZ name and hash
  - TTL is small → why?

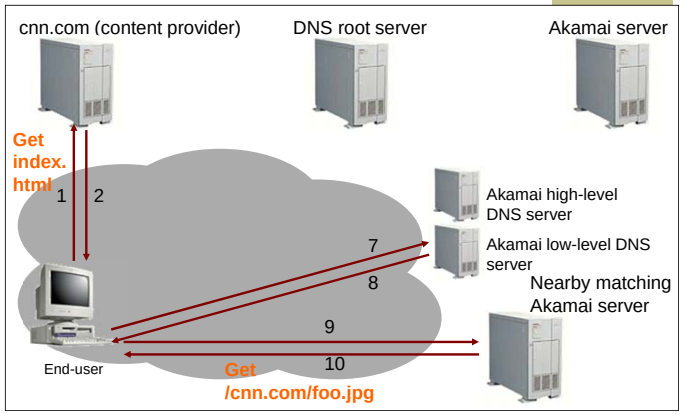
30

How Akamai Works



31

Akamai – Subsequent Requests



32

EdgeSuite – Akamai's Site Delivery Service

Customer CNAME's (aliases)

www.customer.com

- Anyone looking up www.customer.com will be redirected to an Akamai hostname - "customer.d4p.net"
  - No, I do not know why we use "d4p.net". ☺
- customer.d4p.net is CNAME'd to aXXX.g.akamai.net
- Standard Akamai mapping magic sends returns the closest edge server for aXXX.g.akamai.net

EdgeSuite – Akamai's Site Delivery Service

End user never communicates with origin server

- Content retrieved from private hostname, e.g., "origin.customer.com?
- High reliability
  - Thousands of servers backing each other up
  - If one geographic area is disabled, no other area is affected
  - Mitigates some DoS attacks

Uncacheable content is tunneled back to origin

- Can reduce need for tunneling with server-side scripts running on edge caches – now very common

Persistent TCP connections increase performance

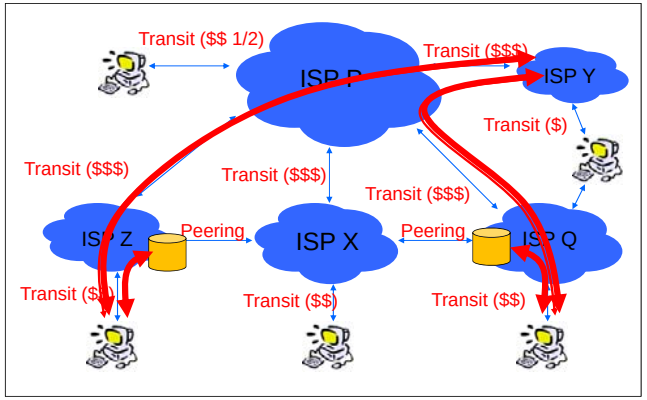
- Helps with downloading of objects to end caches

Overview

- Web
- Consistent hashing
- Peer-to-peer
- CDN
  - Motivation
  - Edge servers
  - Content delivery
  - Mapping
  - Impact on Internet

35

Change in Traffic with CDN Edge Caches



36

Impact of CDN Growth

- Flattening of the Internet
  - More content is served from the edge of the network
  - Also see an increase in peering
  - More traffic remains at edge – reduced load on core
- Changes in the economic relationships
  - Caches benefit users: better performance, reliability
  - Happy customers benefit the CDN
  - ISP benefits since more content is served locally
    - Reduces traffic from provider – direct economic benefit
  - CDNs sometimes place caches in “eye ball” ISPs for free, but economic models change all the time

37

Some Recent Trends

- If CDN’s can deploy caches, why can’t I?
- Content providers have started to deploy CDNs
  - Reduce cost, assuming you are large enough
  - Optimize caching to their specific requiremens
  - Can still use CDNs, e.g., in certain regions, ….
- Internet Service Providers also try deploy CDNs
  - Sometimes difficult to build the business relationships with content owners – too many ISPs!
  - How about the know-how?
  - Hybrid solutions are emerging, e.g., ISPs install hardware and license software from CDNs

39

Summary

- Caching improves web performance
- Caching only at client is only partial solution
  - Not enough locality
- Content Delivery Networks move data closer to user, balance load, increase availability, …
  - Is having impact on structure of the Internet
  - No longer just a solution for static content

40